

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Appellants:	Richard E. KESSLER et al.	§	Confirmation No.:	8193
		§		
Serial No.:	10/677,583	§	Group Art Unit:	2182
		§		
Filed:	October 2, 2003	§	Examiner:	A Hassan
		§		
For:	Scalable Efficient I/O Port	§	Docket No.:	200301779-2
	Protocol	§		

SUPPLEMENTAL APPEAL BRIEF

Mail Stop Appeal Brief – Patents

Date: December 12, 2007

Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

Sir:

Appellants hereby submit this Supplemental Appeal Brief in response to the Notice of Non-Compliant Appeal Brief dated December 12, 2007, for the above-identified application. A Notice of Appeal was electronically filed on September 26, 2007.

TABLE OF CONTENTS

I.	REAL PARTY IN INTEREST	3
II.	RELATED APPEALS AND INTERFERENCES.....	4
III.	STATUS OF THE CLAIMS	5
IV.	STATUS OF THE AMENDMENTS.....	6
V.	SUMMARY OF THE CLAIMED SUBJECT MATTER.....	7
VI.	GROUND OF REJECTION TO BE REVIEWED ON APPEAL.....	9
VII.	ARGUMENT.....	10
	A. Section 102 Rejections over Sharma	10
	1. Claim 3	10
	2. Claim 4	11
	B. Conclusion.....	12
VIII.	CLAIMS APPENDIX.....	13
IX.	EVIDENCE APPENDIX	14
X.	RELATED PROCEEDINGS APPENDIX	15

I. REAL PARTY IN INTEREST

The real party in interest is the Hewlett-Packard Development Company (HPDC), a Texas Limited Partnership, having its principal place of business in Houston, Texas. HPDC is a wholly owned affiliate of Hewlett-Packard Company (HPC). HPC merged with Compaq Computer Corporation (CCC) which owned Compaq Information Technologies Group, L.P. (CITG). In the parent case, Serial No. 09/652,391 (now U.S. Pat. No. 6,738,836), the Assignment from the inventors to CCC was recorded on 08/31/2000, at Reel/Frame 011142/0283. The Assignment from CCC to CITG was recorded on 01/15/2002, at Reel/Frame 012470/0938. The Change of Name document was recorded on 12/02/2003, at Reel/Frame 014177/0428. In the present case, the Change of Name document was recorded on 05/12/2004 at Reel/Frame 014628/0103.

Appl. No. 10/677,583

Supp. Appeal Brief dated December 20, 2007

Reply to Notice of Non-Compliant Appeal Brief of December 12, 2007

II. RELATED APPEALS AND INTERFERENCES

Appellants are unaware of any related appeals or interferences.

Appl. No. 10/677,583

Supp. Appeal Brief dated December 20, 2007

Reply to Notice of Non-Compliant Appeal Brief of December 12, 2007

III. STATUS OF THE CLAIMS

Originally filed claims: 1-9.

Added claims: 10-16.

Claim cancellations: 1-2, 5-16.

Presently pending claims: 3 and 4.

Presently appealed claims: 3 and 4.

Appl. No. 10/677,583

Supp. Appeal Brief dated December 20, 2007

Reply to Notice of Non-Compliant Appeal Brief of December 12, 2007

IV. STATUS OF THE AMENDMENTS

No claims were amended after the final Office action dated July 24, 2007.

V. SUMMARY OF THE CLAIMED SUBJECT MATTER

The various embodiments are directed to a scalable efficient input/output (I/O) port protocol.¹ At least some of the illustrative embodiments are those as defined in claim 3:

3. A distributed multiprocessing computer system, comprising:
a plurality of processors each coupled to an I/O bridge;²
a plurality of memory modules coupled one each to the plurality of processors,³ each memory module able to store data blocks that are shared between said processors,⁴ wherein coherence of the shared data blocks is maintained using a directory based coherence protocol;⁵
at least one I/O device coupled to the I/O bridge,⁶ the I/O device and I/O bridge performs direct memory access (DMA) to transfer data blocks to the memory modules;⁷ and
wherein the I/O bridge obtains exclusive ownership of shared memory module data blocks during I/O device access,⁸ and wherein said I/O bridge includes a DMA device that supports both in-order and out-of-order DMA read and write streams of data blocks.⁹

¹ Specification Title.

² Specification Page 8, lines 1-5; Figure 1, elements 100 and 104.

³ Specification Page 8, lines 1-5; Page 8, lines 19-23; Figure 1, element 102

⁴ Specification Page 9, lines 1-3.

⁵ Specification Page 6, lines 8-10; Page 20, lines 18-22 ("The front-end DIFT 191 performs a number of functions such as managing the processor's directory-based memory coherency...").

⁶ Specification Page 8, lines 15-18; Figure 1, elements 105 and 106

⁷ Specification Page 22, lines 11-14.

⁸ Specification Page 22, line 19 – Page 23, line 9; Page 25, lines 4-6.

⁹ Out-of-order– Specification Page 23, lines 17 – 27; In-order– Page 24, lines 1-12.

Yet still other illustrative embodiments are those as defined in claim 4:

4. The distributed multiprocessing computer system of claim 3 wherein the DMA device that performs an in-order stream of reads of data blocks always receives coherent data blocks that do not have to be written back to the memory module.¹⁰

¹⁰ Specification Page 6, lines 14-18; Page 24, line 17 – Page 25, line 3.

Appl. No. 10/677,583

Supp. Appeal Brief dated December 20, 2007

Reply to Notice of Non-Compliant Appeal Brief of December 12, 2007

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Whether claims 3 and 4 are anticipated under 35 USC § 102(e) by Sharma et al. (U.S. Patent No. 6,647,469, hereafter “Sharma”).

VII. ARGUMENT

A. Section 102 Rejections over Sharma

1. Claim 3

Claims 3 stands rejected as allegedly anticipated by Sharma.

In combination with other limitations, claim 3 specifically recites, “wherein said **I/O bridge** includes a DMA device that supports both in-order and out-of-order DMA read and write streams of data blocks.” The final Office action relies on two portions of Sharma regarding the limitation. First, the final Office action relies on Sharma Col. 7, line 65 – Col. 8, line 9, which states in reference to Sharma’s Figure 3:

Cache unit 315 connects to multiple I/O buses each through a distinct I/O bus interface 320. The I/O bus interface 320 takes a Direct Memory Access (DMA) read request, or DMA write request and forwards it to the appropriate cache unit 315. For example, I/O device 325 connects to the I/O bus interface 330 through the I/O bus 335. The I/O bus interface 330 is connected to cache unit 340 which, in turn, is connected to processor bus interface 305 via internal bus 310 and then to the processor bus 120. From the processor bus 120, a read or write request can be passed to the system memory controller 120 and to main memory 135.¹¹

Noticeably absent is any mention of an ability to support “both in-order and out-of-order DMA read and write streams of data blocks.” Second, the final Office action relies on Sharma Col. 12, line 51 – Col. 13, line 13, which states in reference to Sharma’s Figure 8:

Returning to Step 805, if the data being accessed does represent payload data, but instead consisted of control data, the typical characteristics of such data indicate that only a relatively small chunk of data will be involved in the transfer. Typically, the size of this data is 4 bytes, 8 bytes, or 16 bytes, and the data is typically contained in a single cache line. Step 850 determines if device 150 will access this control data as read only. If the device 150 requires write access to the data, Step 860 initiates traditional DMA read steps necessary to complete the transfer. If, however, the device will only read the data, Step 855 determines if a single transaction can be used to read all of the requested control

¹¹ Sharma Col. 7, line 55 – Col. 8, line 9; Figure 3.

information bytes and if the use of a read current transaction would be beneficial. If the answer to either of these questions is "no" flow continues to Step 860 and traditional DMA read steps are taken. If the control data can be accessed in a single transaction and if the use of the read current would be beneficial, Step 865 determines if an entry is available in the software programmable read current register 705 (FIG. 7). If an entry is available it is used for the read current transaction (similar to Step 820) and if one is not available, Step 870 attempts to set up an appropriate read current register in the I/O interface control traffic dynamically.

Since control data is limited in size, block boundaries are not used with control data. Instead of block boundaries, the information obtained from the software programmable read current register consists of checkup information specific to the control data and the I/O interface readings. This prevents hardware from prefetching data.¹²

Here again, noticeably absent in the quoted portion of Sharma relied upon by the final Office action is any mention of an ability to support "both in-order and out-of-order DMA read and write streams of data blocks." It appears that in only a limited circumstance anything other than "traditional" DMA is used, and in that limited circumstance there is only a single DMA transaction taking data from a register. Appellants respectfully submit that a teaching of a single DMA transaction in limited circumstances fails to expressly or inherently teach a "I/O bridge includes a DMA device that supports both in-order and out-of-order DMA read and write streams of data blocks."

Based on the foregoing, Appellants respectfully submit that the rejection of the claim be reversed, and the claim set for issue.

2. Claim 4

Claim 4 stands rejected as allegedly anticipated by Sharma.

Claim 4 specifically recites, "wherein the DMA device that performs an in-order stream of reads of data blocks always receives coherent data blocks that do not have to be written back to the memory module." The portion of Sharma relied upon by the final Office action¹³ is directed to the content of a DMA request. A

¹² Sharma Col. 12, line 51 – Col. 13, line 13.

¹³ Sharma Col. 9, lines 31-67; Figure 5.

DMA request may be used for many request types.¹⁴ With respect to reads, the cited section states:

If another I/O bus (145 of FIG. 4) had made a previous request for the specific cache line, the I/O bus currently requesting the data may also be allowed access to the data if the ownership is not read current and the cache line is valid.¹⁵

Clearly Sharma contemplates several different read access situations, and thus Appellants respectfully submit that Sharma fails to expressly or inherently teach “wherein the DMA device that performs an in-order stream of reads of data blocks always receives coherent data blocks that do not have to be written back to the memory module.”

Based on the foregoing, Appellants respectfully submit that the rejection of claim 4 be reversed, and the claim set for issue.

B. Conclusion

For the reasons stated above, Appellants respectfully submit that the Examiner erred in rejecting all pending claims. It is believed that no extensions of time or fees are required, beyond those that may otherwise be provided for in documents accompanying this paper. However, in the event that additional extensions of time are necessary to allow consideration of this paper, such extensions are hereby petitioned under 37 C.F.R. § 1.136(a), and any fees required (including fees for net addition of claims) are hereby authorized to be charged to Hewlett-Packard Development Company's Deposit Account No. 08-2025.

Respectfully submitted,

/Mark E. Scott/

Mark E. Scott
PTO Reg. No. 43,100
CONLEY ROSE, P.C.
(512) 391-1900 (Phone)
(512) 320-9181 (Fax)
ATTORNEY FOR APPELLANTS

HEWLETT-PACKARD COMPANY
Intellectual Property Administration
Legal Dept., M/S 35
P.O. Box 272400
Fort Collins, CO 80527-2400

¹⁴ For example, block 520 identifies the request type as read or write. Sharma Col. 9, lines 57-59.

¹⁵ Sharma, Col. 9, lines 42-46.

VIII. CLAIMS APPENDIX

1.-2. (Cancelled).

3. (Previously presented) A distributed multiprocessing computer system, comprising:

a plurality of processors each coupled to an I/O bridge;

a plurality of memory modules coupled one each to the plurality of processors, each memory module able to store data blocks that are shared between said processors, wherein coherence of the shared data blocks is maintained using a directory based coherence protocol;

at least one I/O device coupled to the I/O bridge, the I/O device and I/O bridge performs direct memory access (DMA) to transfer data blocks to the memory modules; and

wherein the I/O bridge obtains exclusive ownership of shared memory module data blocks during I/O device access, and wherein said I/O bridge includes a DMA device that supports both in-order and out-of-order DMA read and write streams of data blocks.

4. (Original) The distributed multiprocessing computer system of claim 3 wherein the DMA device that performs an in-order stream of reads of data blocks always receives coherent data blocks that do not have to be written back to the memory module.

5.-16. (Cancelled).

Appl. No. 10/677,583

Supp. Appeal Brief dated December 20, 2007

Reply to Notice of Non-Compliant Appeal Brief of December 12, 2007

IX. EVIDENCE APPENDIX

None.

Appl. No. 10/677,583

Supp. Appeal Brief dated December 20, 2007

Reply to Notice of Non-Compliant Appeal Brief of December 12, 2007

X. RELATED PROCEEDINGS APPENDIX

None.